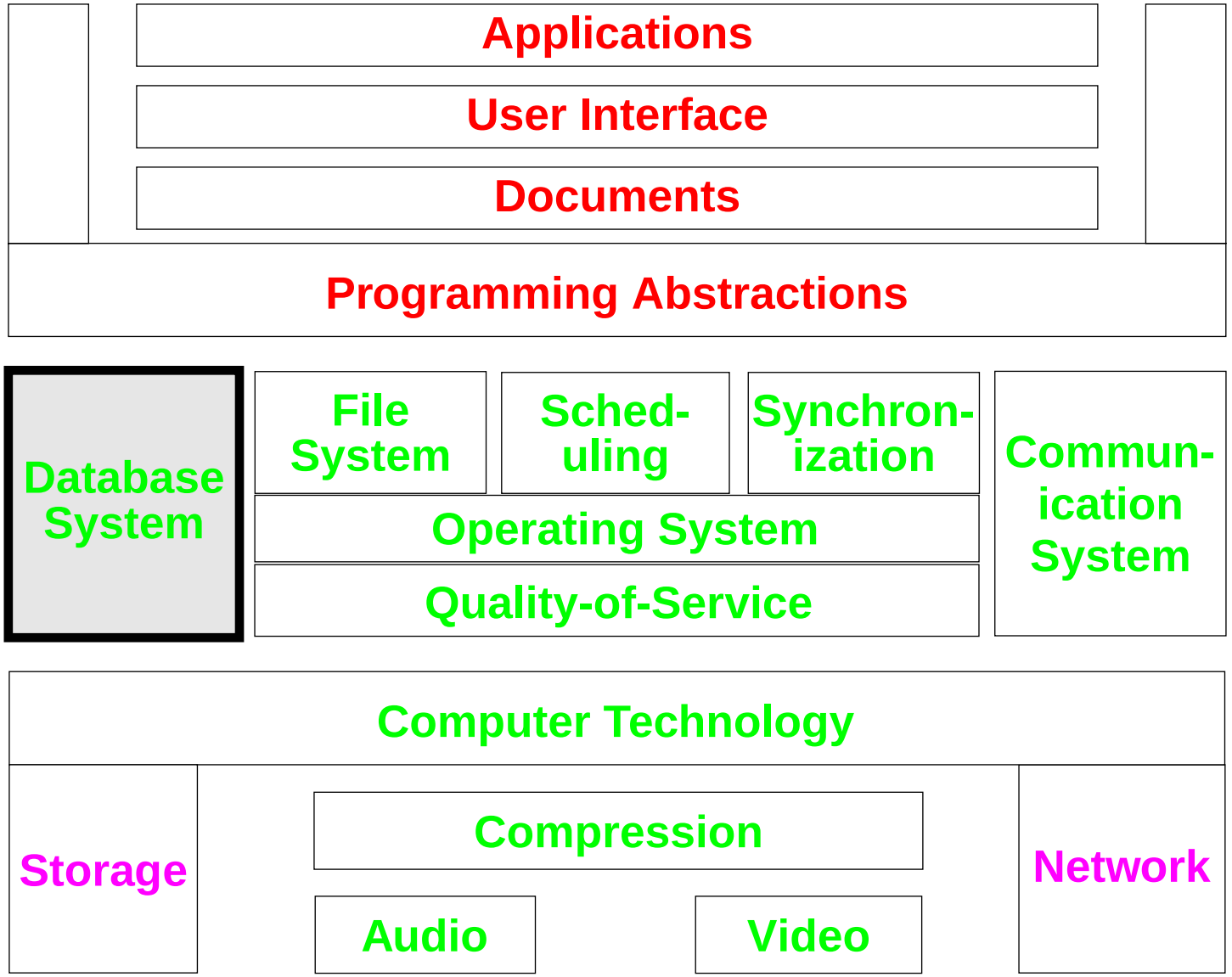
Multimedia Systems: Database Support

Ralf Steinmetz
Lars Wolf

Darmstadt University of Technology
Industrial Process and System Communications

Merckstraße 25 – 64283 Darmstadt – Germany


Applications
User Interface
Documents
Programming Abstractions
Database System
File System
Sched-uling
Synchron-ization
Operating System
Quality-of-Service
Commun-ication System
Computer Technology
Storage
Compression
Audio
Video
Network

Contents

1. Need for Databases: Motivation

Data bases used to

- Provide structured data storage
- Allow efficient search and retrieval

Classic properties of databases:

- Data independence (data abstraction)
- Application neutrality (openness)
- Multi-user operation (concurrency control)
- Fault tolerance (transactions, recovery)
- Access control

⇒ extend these services to cover multimedia data
and store all kinds of data which can be digitally stored

Requirements of Multimedia Databases

Due to characteristics of multimedia:

- Management of large amount of data
- Relations between data of different media
- Real-time access
- Provision of multimedia-specific search methods
- Long-lasting transactions

2. Multimedia Database Architecture

View at database

- related to concepts
- comprises components

Data model

Transaction model

Storage model

Query model & Interface model

Data Model

Here

- generic overview (not part of all MM Databases)

1. Level: data types

- basic types
- e.g. image, real, sound, compound video, speech, ...

2. Level: objects (classes)

- includes attributes and methods (behaviour)
- attributes: application dependent, generic for many objects
- e.g. audio, real-time, collection, spatial, fuzzy set, .

3.Level: relationship among objects (classes)

- type & value with graphical representation
- e.g. behind, before, ..., at, after, ...,

Transaction Model

Scope

- to define creation, modification, delete of data
- works with a “manager” at each layer

1.Layer: concurrency control

- Pessimistic locking: exclusive locking used, because conflicts are expected and no recovery would be possible, short term check out
- Optimistic locking: nonexclusive locking used, because no conflicts are expected or recovery is possible, long term check out

2..Layer: lock mechanisms

- type (tokens, semaphore, ..), granularity (entire object, part of it,..)

3.Layer: update Management

- commit and/or abort updates

4.Layer: version Control

5.Layer: integrity Check

Storage Model

Scope

- access to various storage devices

1. Access control

- interface to directory/ies
- knowledge how to store/access information

2. Index Management

- index mechanisms depends on data layer and type

3. Buffer Management

- e.g. static vs. dynamic partitioning

4. I/O Management

- handling of multimedia file system
- page sizes, etc.

Query & Interface Model

Queries

- Scope
  - how to specify data contents
- Compound Query
- Subquery & Query Evaluation Plan
- Query engines

Multimodal interface layers

- scope
  - how to interface & display information
- collection of query interfaces
  - dedicated to special media types
- query refinement
  - thesaurus & context based access
- query interator
  - combine all individual results to single query

3. Issues Related to Media Data

Large Volume of Data

Optimizations for data storage:

- Raw device access can be used to exploit full capacity
- Efficient connection with communications subsystem enables more transparent spooling of isochronous data
- OODB, e.g., allows fields to have arbitrarily large size

Optimizations for retrieval

- Real-time access is often needed
- Prejoined relations can speed answers to queries
- Indexes used to quickly locate data

Multimedia Data – Binary Large Objects

Characteristics:

- Variable size
- Structure of data is unknown to database system
- “Read bits” and “write bits” are only operations available to database client
- Interpretation and manipulation are responsibility of client

Typical uses:

- Executable code
- CAD file
- Client- or application- specific data

Media Data - OO Structure

Sub- and super- types:

- Properties and attributes inherited from supertype(s)
- Multiple inheritance possible
- Types arranged in lattice:
  - All types inherit from ‘object’ supertype and
  - ‘Bottom’ subtype inherits from all types

Part-of relationships:

- Attribute of object may itself be a structured object
- Enables hierarchical composition and levels of abstraction

Object reference:

- Enables object sharing
- Same image might be shared in slide show, for example

Media Data as Imprecise Model

Data characteristics

- data models real world
- however, it is **impresice & incomplete** representation (audio, video, image, ..)

Implications

- to handle impresice & inconplete representations/models
- to cope with incomplete & inprecise quieries

Examples

- .. is similar to ..
- i.e., fuzzy like behaviour

Multimedia Data Objects

Multimedia data objects consist of:

- Raw data
  - unformatted sequence of symbols (characters, pixels, audio samples,...)
  - potentially large binary objects (“BLOBs”)
- Type information
  - needed for correct data interpretation
- Descriptive data
  - optional information about the data
- Meta-data for associations between related objects
  - synchronize objects, share parts of objects

Database requirements:

- Isochronous transfer from storage device
- Synchrony between audio and video streams
- Large amount of storage

Multimedia Data Objects – Examples

Image:

- Raw – pixels
- Registration – height, width, color table, coding method (JPEG, ...)
- Description – lines, areas, objects, situations (e.g., birthday party)

Image sequence:

- Raw – pixels
- Registration – frames/s, coding information
- Description – information about scene (e.g. monkeys in zoo)
- Meta – synchronization points

4. Issues Related to Transactions

Role of Time

Transactions on multimedia data may be long-lasting

- Locking strategy must not exclude access to other clients, as is typical in traditional DBMSs
- “Dirty reads” should be permitted
- Rollback may not be required

Time as predicate

- Time information needed to assemble ropes / strands
- Editing of multimedia data may require fast random access based upon timestamps

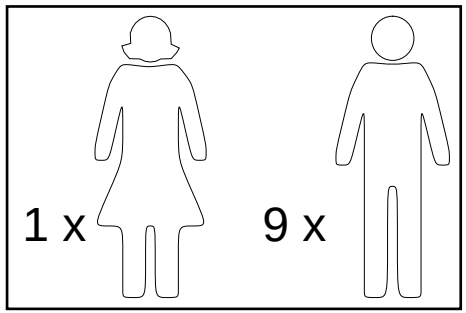
5. Issues Related to Query and Interface

Multimedia Database Interface: Browsing

Interactive search:

- Icons (pictorial representations, possibly created from real data)
- Miniatures (items themselves in reduced form - difficult for AV media)
- Attribute values (textual form)

Example:



Group of 10 persons (1 woman, 9 men)

Query Languages

Conventional

- SQL like very good for structured text

Approaches for other media

- attribute based:
  textual description of information in other media
- contents based:
  description in the media itself

Syntax of language

- graphics: most often visual objects used to express contents
- video: making use of images, not yet movements
- audio: hard to compare in audio domain

Attribute-Based Queries

Search multimedia objects by their attribute values

Features:

- Makes use of annotations to multimedia objects
- Normally ad-hoc
- Simple Boolean combinations of comparisons
- Does not require understanding of contents of objects

Examples:

- Find all video snips authored by Marilyn
- Locate the sound effect that accompanies text *T*
- Find all images more than 1 year old

Content-Based Queries

Search multimedia objects by their contents

Features:

- Powerful but in general too difficult to realize
- “Confidence level” of result depends upon semantic understanding
- May require large amount of computation

Examples:

- Find all trapezoids in graphic *G*
- Find all photos showing ten people:

- Find the loudest portion of Klee’s Concerto in H

Multimedia Query Examples

Graphic query:

- CAD DB: retrieve all drawings that include object with specific shape

Image query:

- Banking DB: retrieve checks with certain signature

Video query:

- Video DB: retrieve videos with a specified actor

Audio query:

- Radio DB: retrieve talk shows discussing car driving speed limits

Note:

- Use of textual (attribute, value) pairs is commonly used today

6. Summary

Databases have to support:

- Real-time characteristics and
- Large size of of audio and video data

New data structuring methods are necessary

Queries are more difficult:

- Departure from traditional text based interface
- Media content should be expressable